

LISTING OF THE CLAIMS

Please consider the claims as follows:

1. (Previously presented) The method of claim 5, further comprising:

calculating a helper hotness rating at each of a plurality of helper servers (HSs) for each of a plurality of SM objects that are hosted at a content server connected to the HSs in a network, each helper hotness rating being a total number of client requests for a particular SM object divided by a time period during which the client requests are received;

calculating a server hotness rating at the content server for each SM object, each server hotness rating for the particular SM object being a sum of the calculated helper hotness ratings received from all of the HSs;

categorizing each SM object into one of a plurality of server hotness categories based on the calculated server hotness rating; and

pushing each SM object from the content server to a fraction of HSs, each fraction being determined according to the server hotness category, the HSs caching the fractions of each SM object for distribution to a plurality of clients, the HSs being interposed between the content server and the clients.

- 2.-4. (Cancelled)

5. (Currently amended) A method for caching streaming multimedia (SM), comprising:

calculating a helper hotness rating at a helper server (HS) for each of a plurality of SM objects that are hosted at a content server connected to the HS in a network, each helper hotness rating being a total number of client requests for a particular SM object divided by a time period during which the client requests are received;

categorizing each SM object into one of a plurality of helper hotness categories based on the calculated helper hotness rating; and

pulling a fraction of the SM objects by the HS from the content server, said

fraction being determined according to the helper hotness category, the HS caching the fractions of each SM object for distribution to a plurality of clients, the HS being interposed between the content server and the clients.

6. (Previously presented) The method as recited in claim 5, wherein pulling by said HS is in response to client requests.

7. (Previously presented) The method as recited in claim 5, wherein a deterministic cache placement and replacement policy is implemented at the HSs.

8. (Previously presented) The method of claim 5, wherein a random cache placement and replacement policy is implemented at the HSs.

9-13. (Cancelled).

14. (Withdrawn) A method for storing a streaming multimedia (SM) object in a network having a content server which hosts SM objects for distribution over said network through a plurality of helper servers (HS) to a plurality of clients, said streaming media object being comprised of a plurality of successive time-ordered chunks, wherein a chunk is further comprised of an integer multiple number of HS storage disk blocks, said method comprising:

- i) receiving a streaming media object;
- ii) determining whether sufficient disk space is available on said at least one of said plurality of HSs to store said received SM object;
- iii) storing said received SM object at said at least one of said plurality of HSs if it is determined that there is sufficient disk space; and
- iv) performing the following steps, if it is determined that there is insufficient disk space available to store the received SM object:
 - a) identifying at least one SM object from among a plurality of SM objects hosted by said HS which is not in use and has an access time which is

least recent, wherein said access time corresponds to a time when said SM object was last requested; and

b) replacing chunks of said identified at least one SM object corresponding to said received SM object to store at least a portion of said received SM object.

15. (Withdrawn) A method as recited in claim 14, wherein the step of replacing chunks further comprises the step of replacing the chunks having the largest associated time-stamp value of said at least one identified SM object.

16. (Withdrawn) A method as recited in claim 15, wherein the step of replacing said chunks having the largest time-stamp value further comprises the step of replacing said chunks having the largest time-stamp value on a disk block by disk block basis starting from the end of the chunk.

17. (Withdrawn) A method for storing a streaming media (SM) object in a network having a content server which hosts SM objects for distribution over said network through a plurality of helper servers (HSs) to a plurality of clients, said SM object being comprised of a plurality of successive time-ordered chunks, wherein a chunk is further comprised of some integer multiple number of disk blocks, said method comprising:

- i) receiving said SM object;
- ii) determining whether there is disk space available on said one of said plurality of HSs;
- iii) storing said SM object at said at least one HS if it is determined that there is sufficient disk space available; and
- iv) performing the following steps, if it is determined that there is insufficient disk space available:

a) composing a set of SM objects from among a plurality of SM objects stored on said disk space whose access time is determined to be least recent, where said access time corresponds to a time when said SM object was

last requested; and

b) replacing chunks of said SM objects belonging to said composed set with chunks of said received SM object.

18. (Withdrawn) A method as recited in claim 17, wherein the replacing step further comprises the step of replacing a chunk having an associated highest time-stamp value from each of said SM objects belonging to said composed set in a round-robin basis.

19. (Withdrawn) A method as recited in claim 17, wherein said composed set is formed by including only SM objects having a helper hotness rating below a predefined threshold.

20. (Withdrawn) A method for storing a streaming media (SM) object in a network having a content server which hosts SM objects for distribution over said network through a plurality of helper servers (HSs) to a plurality of clients, said SM object being comprised of a plurality of successive time-ordered chunks, wherein a chunk is further comprised of some integer multiple number of disk blocks, said method comprising:

- i) receiving said SM object at one of said plurality of HSs in said network; and
- ii) randomly determining a fraction and storing said fraction of said SM object at said one of said HSs in the network.

21. (Withdrawn) A method as recited in claim 20, wherein the step of randomly determining said fraction said fraction comprises the steps of:

- i) estimating at said one of said plurality of HSs, the number of HSs in said network, K;
- ii) calculating a value, P, that is inversely proportional to the value K;
- iii) generating a random value, R, for a received chunk of said received SM object;
- iv) comparing said generated random value, R, with said calculated value P;

v) caching the received chunk, if it is determined that the generated random value, R , is less than the calculated value P ; and

vi) discarding the received chunk, if it is determined that the generated random value, R , is greater than the calculated value P .

22. (Withdrawn) The method according to claim 21, wherein the step of generating the random value R is performed by utilizing the IP address of said one of said plurality of HSs as a seed.

23. (Withdrawn) A method for storing a streaming media (SM) object in a network having a content server which hosts SM objects for distribution over said network through a plurality of helper servers (HSs) to a plurality of clients, said SM object being comprised of a plurality of successive time-ordered chunks, wherein a chunk is further comprised of some integer multiple number of disk blocks, said method comprising:

- i) requesting said SM object at one of said plurality of HSs in said network;
 - ii) receiving said SM object at one of said plurality of HSs in said network;
- and
- iii) randomly determining a fraction and storing said fraction of said SM object at said one of said HSs in the network.

24. (Withdrawn) A method as recited in claim 23, wherein the step of randomly determining said fraction said fraction comprises the steps of:

- i) estimating at said one of said plurality of HSs, the number of HSs in said network, K ;
- ii) calculating a value, P , that is inversely proportional to the value K ;
- iii) generating a random value, r , for a received chunk of said received SM object;
- iv) comparing said generated random value, R , with said calculated value P ;
- v) caching the received chunk, if it is determined that the generated random value, R , is less than the calculated value P ; and

vi) discarding the received chunk, if it is determined that the generated random value, R, is greater than the calculated value P.

25. (Withdrawn) The method according to claim 24, wherein the step of generating the random value R is performed by utilizing an arrival time as the seed of said SM object at said one of said plurality of HSs.

26. (Withdrawn) A method for storing a streaming media (SM) object in a network having a content server which hosts SM objects for distribution over said network through a plurality of HSs to a plurality of clients, said SM object being comprised of a plurality of successive time-ordered chunks, wherein a chunk is further comprised of some integer multiple number of disk blocks, said method comprising:

- i) receiving said SM object at one of said plurality of HSs;
- ii) determining whether there is disk space available on said one of said plurality of HSs to store said SM object;
- iii) storing said SM object at said one of said plurality of HSs if it is determined that there is sufficient disk space available; and
- iv) performing the following steps, if it is determined at the determining step that there is insufficient disk space available:

- a) composing a set of streaming media objects from among a plurality of streaming media objects stored on said disk space, whose access time is determined to be above a predetermined threshold, wherein said access time is a time when said SM object was last retrieved;

- b) assigning a unique integer value to each chunk of the composed set of SM objects based at least one of a name of the SM object, a time-stamp of the chunk, and a last access time of the chunk;

- c) generating a random value, R, having a range between one and the largest unique integer value assigned; and

- d) replacing a stored chunk having said assigned integer value which corresponds to the generated random value with a chunk of said received SM

object.

27. (Withdrawn) A method for servicing a client request for a streaming media object in a network having a content server which hosts SM objects for distribution over said network through a plurality of HSs to a plurality of clients, said SM object being comprised of a plurality of successive time-ordered chunks, said method comprising:

- i) receiving a request for at least a portion of an SM object at a local HS, said request including a playback starting time;
- ii) determining whether a chunk having a starting time equal to the requested playback starting time resides on a disk associated with said local HS;
- iii) delivering to one of said plurality of clients said one or more chunks having an associated starting time equal to the requested playback starting time from said local HS and updating said playback starting time;
- iv) identifying a server in said network storing at least M chunks which have a starting time equal to said updated playback starting time and which satisfies a minimum cost criterion, if it is determined that said updated playback starting time is not equal to an end-time of said SM object;
- v.) receiving said at least M chunks from said identified server at said local HS until it is determined that a next chunk to be retrieved is stored at said local HS;

28. (Withdrawn) A method for determining an optimal sequence of cache accesses to service a request for a streaming media (SM) object in a network including a content server which hosts SM objects for distribution over said network through a plurality of helper servers (HSs) to a plurality of clients, the SM objects being comprised of a plurality of successive time-ordered chunks, the method comprising the steps of:

- (i) receiving a request at a local HS for an SM object, said request including a requested playback starting time;
- (ii) determining whether a chunk of an SM object having a starting time equal to the requested playback starting time is stored at said local HS;
- (iii) retrieving said one or more chunks of said SM object when step (ii) is

satisfied and updating the requested playback starting time;

(iv) identifying a minimum cost server in the network which stores at least M chunks of said SM object having a starting time equal to the updated playback starting time;

(v) retrieving said at least M chunks on a chunk by chunk basis from said identified minimum cost server and said local HS;

(vi) retrieving said next chunk from said local HS and updating the requested playback starting time;

(vii) repeating step (ii) through step (vi) until the entire streaming media object has been retrieved.

29. (Withdrawn) The method of claim 28, wherein at step (v), a chunk is always retrieved from said local HS if it is available.

30. (Withdrawn) The method of claim 29 further comprising the step of returning chunks of said requested SM object to one of said plurality of clients as they are retrieved at said local HS.

31.-32. (Cancelled).

33. (Previously presented) The method as recited in claim 1, wherein pushing by said content server is in response to client requests.

34. (Previously presented) The method as recited in claim 1, wherein a deterministic cache placement and replacement policy is implemented at the HSs.

35. (Previously presented) The method of claim 1, wherein a random cache placement and replacement policy is implemented at the HSs.